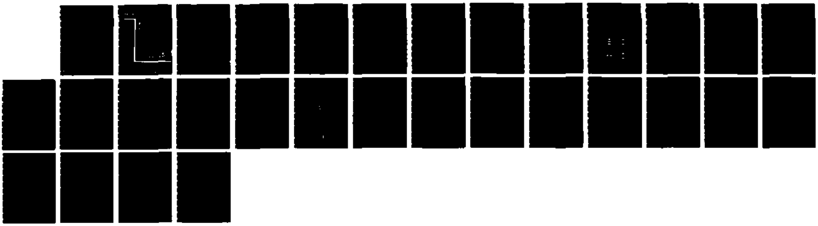
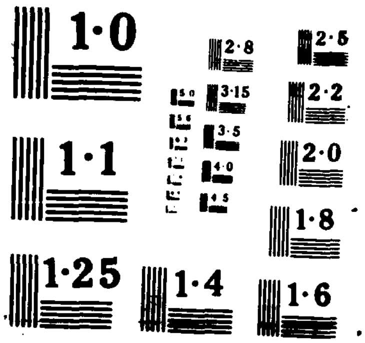
AD-A176 514 FAULT-TOLERANT SYSTEM ANALYSIS: IMPERFECT SWITCHING AND 1/1
MAINTENANCE(U) ANALYTIC SCIENCES CORP READING MA
M H VEATCH ET AL. JAN 87 AFHRL-TP-86-49
UNCLASSIFIED F33615-82-C-0002 F/G 9/5 NL




1·0
2·8
2·5
3·15
2·2
3·5
1·1
2·0
4·0
4·5
1·8
1·25
1·4
1·6

12

AFHRL-TP-86-49

AIR FORCE HUMAN RESOURCES LABORATORY

FAULT-TOLERANT SYSTEM ANALYSIS: IMPERFECT SWITCHING AND MAINTENANCE

Michael H. Veatch
Robert D. Foley

The Analytic Sciences Corporation
One Jacob Way
Reading, Massachusetts 01867

LOGISTICS AND HUMAN FACTORS DIVISION
Wright-Patterson Air Force Base, Ohio 45433-6503

January 1987

Final Technical Paper for Period June 1985 - June 1986

Approved for public release; distribution is unlimited.

AIR FORCE SYSTEMS COMMAND
BROOKS AIR FORCE BASE, TEXAS 78235-5601

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
Unclassified

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE

3. DISTRIBUTION / AVAILABILITY OF REPORT
Approved for public release; distribution is unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

5. MONITORING ORGANIZATION REPORT NUMBER(S)
AFHRL-TP-86-49

6a. NAME OF PERFORMING ORGANIZATION
The Analytic Sciences Corporation

6b. OFFICE SYMBOL *(If applicable)*

7a. NAME OF MONITORING ORGANIZATION
Logistics and Human Factors Division

6c. ADDRESS *(City, State, and ZIP Code)*
One Jacob Way
Reading, Massachusetts 01867

7b. ADDRESS *(City, State, and ZIP Code)*
Air Force Human Resources Laboratory
Wright-Patterson Air Force Base, Ohio 45433-6503

8a. NAME OF FUNDING / SPONSORING ORGANIZATION
Air Force Human Resources Laboratory

8b. OFFICE SYMBOL *(If applicable)*
HQ AFHRL

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
F33615-82-C-0002

8c. ADDRESS *(City, State, and ZIP Code)*
Brooks Air Force Base, Texas 78235-5601

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
62205F	1710	00	26

11. TITLE *(Include Security Classification)*
Fault-Tolerant System Analysis: Imperfect Switching and Maintenance

12. PERSONAL AUTHOR(S)
Veatch, M.H.; Foley, R.D.

13a. TYPE OF REPORT
Final

13b. TIME COVERED
FROM Jun 85 TO Jun 86

14. DATE OF REPORT *(Year, Month, Day)*
January 1987

15. PAGE COUNT
32

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP
15	05	
14	04	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
communication
fault-tolerant avionics
identification
logistics analysis
mean time between critical failure
mean time between failure (Continued)

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

This final report presents the results of research into important areas of concern for fault-tolerant avionics systems: testability analysis and innovative repair policies. A method of quantifying the effects of undetected errors and false alarms on system reliability has been developed and included in the Mission Reliability Model (MIREM). Innovative repair policies, such as repair at a degraded level, were identified and added to the model.

MIREM is a computer program designed to evaluate the reliability and operating capability of advanced electronics in the early stages of design. The model is applicable to integrated systems that achieve fault tolerance through dynamic reconfiguration, fault detection, and resource sharing.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Nancy A. Perrigo, Chief, STINFO Office

22b. TELEPHONE *(Include Area Code)*
(512) 536-3877

22c. OFFICE SYMBOL
AFHRL/TSR

83 APR edition may be used until exhausted.
All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

AFHRL Technical Paper 86-49 January 1987

FAULT-TOLERANT SYSTEM ANALYSIS:
IMPERFECT SWITCHING AND MAINTENANCE

Michael H. Veatch
Robert D. Foley

The Analytic Sciences Corporation
One Jacob Way
Reading, Massachusetts 01867

LOGISTICS AND HUMAN FACTORS DIVISION
Wright-Patterson Air Force Base, Ohio 45433-6503

Reviewed by

Lee H. Dayton, 1st Lt, USAF
Logistics Systems Branch
Logistics and Human Factors Division

Submitted for publication by

Joseph F. Nerad, Major, USAF
Chief, Logistics Systems Branch
Logistics and Human Factors Division

SUMMARY

This final report presents the results of research into two important areas of concern for fault-tolerant avionics systems: testability analysis and innovative repair policies. The algorithms developed from this research have been included in the Mission Reliability Model (MIREM) and verified by comparison with known results from several Integrated Communication, Navigation, and Identification Avionics architectures.

The purpose of the testability analysis was to develop techniques for assessing the impact of imperfect switching on the overall reliability of fault-tolerant avionics. A method of quantifying the effects of undetected errors and false alarms has been developed and included in MIREM. Under the next phase of the program, three repair statistics were identified: Mean Time To Repair, Mean Time Between Maintenance Actions, and Inherent Availability. These were used to define four alternative repair policies: immediate repair, deferred repair, scheduled maintenance, and repair at degraded level. Also included in MIREM as model outputs, these four options offer greater flexibility in evaluating and developing avionics designs.

Conclusions are given, along with recommendations for use of MIREM in the Integrated Maintenance Information System. As a result of the enhancements to MIREM, the model now has the added capability to be used as a predictor of performance during testing, rather than solely as a tradeoff and evaluation tool.

PREFACE

This report presents the algorithms and other conclusions of the Fault-Tolerant System Analysis effort. The research covers the areas of testability analysis and innovative maintenance policies for fault-tolerant systems. The testability analysis task was performed under subcontract by Dr. Robert Foley of the Georgia Institute of Technology. This work is sponsored by the Air Force Human Resources Laboratory. The guidance and support of Lt Lee Dayton of this Laboratory are greatly appreciated.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

FAULT-TOLERANT SYSTEM ANALYSIS:
IMPERFECT SWITCHING AND MAINTENANCE

1. INTRODUCTION

Recent trends toward integration and fault tolerance in avionics have created a need for new reliability analysis techniques that capture these characteristics and can identify support concepts that exploit the fault-tolerant nature of these systems. One archetypal fault-tolerant system, the Integrated Communication, Navigation and Identification Avionics (ICNIA), is being designed with dynamic reconfiguration that allocates common system resources to a variety of radio functions across a wide spectrum of frequencies. Dynamic reconfiguration will allow faults to be managed and resources to be effectively shared between required functions.

Another motivation for research into analysis techniques is that, historically, logistics engineering disciplines have been applied to avionics in the later stages of development. To ensure that advanced avionics are reliable and supportable, logistics engineering techniques are needed that can be implemented early in the development cycle, before the design is fixed.

The Mission Reliability Model (MIREM) was developed to help meet these needs. The Fault-Tolerant Systems Analysis program was conducted to extend the MIREM concept to address logistics engineering issues encountered further into the development cycle and to broaden the applicability of MIREM. Two specific areas of investigation were identified by the Air Force and the ICNIA development contractors as particularly relevant for advanced systems:

1. Testability Analysis: Develop techniques for assessing the impact of imperfect switching on the overall reliability of fault-tolerant avionics.

2. Innovative Repair Policies: Investigate innovative repair policies for fault-tolerant systems and quantify their impact on reliability and availability.

The algorithms developed in these two areas provided the technical basis for a new version of MIREM (MIREM3), which is documented in Veatch and Gates (1986) and has been installed at the Aeronautical Systems Division Computer Center, Wright-Patterson AFB, Ohio, on a VAX 11/780.

Chapter 2 summarizes the Testability Analysis as performed by Dr. Foley and abstracted by TASC. The derivation of these results, taken from Foley and Suresh (1986) with minor editing to enhance clarity, is presented in Appendix A. Note that the imperfect switching reliability algorithms derived here differ somewhat from those implemented in MIREM3. Chapter 3 describes the Innovative Repair Policies results. Conclusions and recommendations are presented in Chapter 4.

2. TESTABILITY ANALYSIS

To design a fault-tolerant system properly, design engineers need quantitative information on the performance of various prototype systems. MIREM allows design engineers to determine the change in reliability due to the changes in the system design. A simple example of the kind of structure analyzed by MIREM is illustrated in Figure 1.

At the lowest level, pools of interchangeable system resources are identified. Branches are alternate, identical paths within a pool, each containing one or more resources in series. In general, several different functions must be performed by the system. Each function utilizes a certain number of branches (or fractions of a branch) in a pool. The combined resource requirement for a set of required functions depends on a number of timing issues. Given a total resource requirement of k, a pool with n parallel branches is evaluated as a k-of-n structure. Reliability for a set of series pools, called a chain, is the

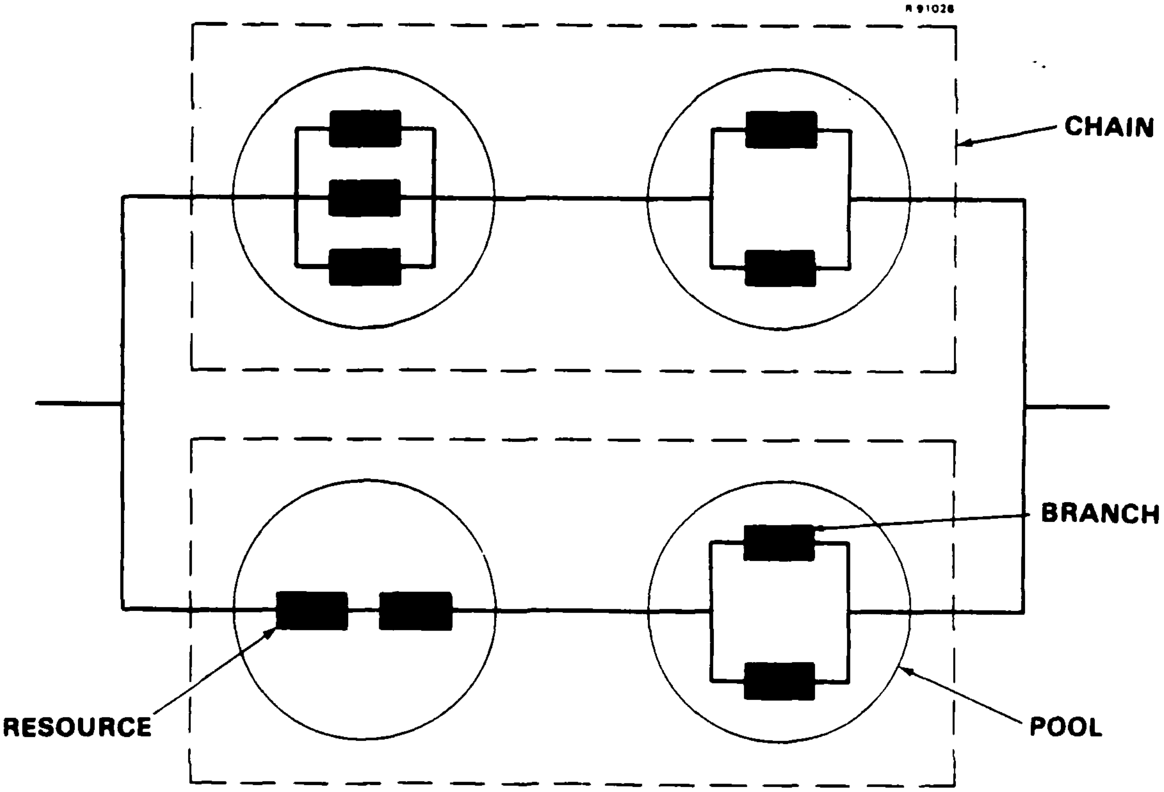


Figure 1. A Two-Level System Structure.

product of the probabilities of each pool having sufficient resources operating.

At a higher level, functions can be allocated between parallel chains. A chain is a set of pools that is switched (reconfigured) as a group. In many cases, a chain will correspond to a Line Replaceable Unit (LRU) because LRUs have separate power supplies and limited inter-LRU connections. A set of functions is available on parallel chains if there is an allocation of functions to chains such that each chain can support its allocated functions.

Previous work with MIREM has not taken into account undetected errors or false alarms. MIREM assumes that the internal system monitor knows for certain whether each component is working or failed. In reality, the monitor may mistakenly believe that a particular component is broken when it is not, or that it is working when it is actually broken. In total, there are four possible combinations of the believed state and actual state of the component. There are three possible actual states of the system: all critical functions are being supported, all critical functions a~~r~~e not being supported (but can be) due to an incorrect configuration, and all critical functions cannot be supported. The two believed states of the system, all critical functions are (are not) being supported, give six combinations of system states. For the purpose of discussion, some of these states will be combined to give four system states:

A. All critical functions are being supported, and the system monitor believes that all critical functions are being supported;

B. All critical functions cannot be supported, and the monitor believes that all critical functions are not being supported;

C. All critical functions can be supported, but the monitor believes that all critical functions are not being supported;

D. All critical functions are not being supported, but the monitor believes that all critical functions are being supported.

Clearly, state A is the preferred state. State B is caused by the occurrence of one or more detectable errors. State C is caused by false alarms. State C represents lost opportunity in that the mission would most likely be prematurely aborted if the monitor believes the system is down when actually it is capable of functioning. State D, which is caused by nondetected errors, seems particularly undesirable. In state D, the monitor believes that all critical functions are supported when they are not. State D might result in a mission's being continued even though the mission is doomed to failure because some of the critical functions are not supported. State D is the state most likely to result in loss of aircraft and crewmen.

As mentioned earlier, MIREM previously assumed a perfect monitor, a monitor which detects all failures and makes no false alarms. We will replace this assumption with the assumption that the monitor is imperfect; the monitor may not realize that some components have failed, and the monitor may incorrectly believe that other components have failed.

2.1 Classification of Mission Outcome

We will classify any mission into one of four possible categories:

1. Mission Success (1,1): The mission was successful and the monitor believed the mission was successful.

2. False Abort (1,0): The mission was aborted when it should not have been.

3. Unknown Mission Failure (0,1): A critical failure occurred but the mission was not aborted because the monitor was not aware of the critical failure.

4. Correct Abort (0,0): The mission was aborted when it should have been.

The principal quantities of interest are the probabilities that a mission will fall into each of the four categories and the mean time in the state where the system is up and the monitor believes the system is up, denoted by E[T]. These quantities of interest cannot be computed exactly since the algorithm for allocating functions is not completely known. However, algorithms are developed in Appendix A to compute the upper and lower bounds for all of these quantities.

2.2 Implementation and Numerical Examples

Implementation. The Appendix A algorithms were implemented in Fortran-77 and run on an IBM 4381 at the Georgia Institute of Technology. Double precision was used throughout. The following notation is used for the testability-related parameters:

λ_i is the failure rate on branch i

p_i is the probability of detecting a failure on branch i

α_i is the rate of false alarms for branch i

t_m is the length of the mission

For simplicity, assume that p_i is the same for all branches i. The algorithms developed in Appendix A are used to bound the probability of each mission outcome. Bounds on E[T] are computed by numerically integrating the mission success probability bounds using the standard MIREM algorithm from Veatch and Gates (1986).

A variety of test cases have been used to validate the implementation of the algorithms and to explore the implications of the testability parameters. Example 1 was constructed as a simple illustration. Example 2 is a standard MIREM test case that has been used in the literature. The results are presented below.

Example 1. The system consists of one pool containing two branches. The single critical function requires one branch. Table 1 gives the results for the parameters $t_m = 3$ hours, $\lambda_i = -\frac{1}{6} \ln 0.9$, $p_i = 0.5$, $\alpha_i = \lambda_i$. The true values are represented by the actual column and were computed manually for this system.

Example 2. This example (Figure 2) is taken from Veatch and Calvo (1983). It also was analyzed in Foley and Suresh (1984), and is used in Veatch and Gates (1986), but with different testability parameters. Several variations of this example have been created. In the version discussed here, the Global Positioning System (GPS) function cannot use chain 3 (Digital B), GPS requires two preprocessors, the total failure rate is 2230×10^{-6} hours^{-1}, and the power supplies are necessary to use any pool in their chain. Table 2 gives the reliability results for the parameters $t_m = 3$ hours, $p_i = 0.5$, and $\alpha_i = \lambda_i$. The bounds for E[T] are 508.3 hours and 593.91 hours.

The algorithm was also tested on all examples by setting $\alpha_i = 0$ and $p_i = 1.0$ (no nondetected failures or false alarms). The results matched with the perfect monitor results obtained by Foley and Suresh (1984) and in each case, the upper and lower bounds differed only in the sixth decimal place. Note that in this case there are only two possible outcomes: mission success and correct abort. The program was then run with $\lambda_i = 0$ and α_i set to the original λ_i (failures replaced by false alarms). Note

Table 1. Example 1 Testability Results

Outcome	Lower bound	Actual	Upper bound
Correct Abort	0.652×10^{-3}	0.664×10^{-3}	0.685×10^{-3}
Unknown Mission Failure	0.255×10^{-1}	0.258×10^{-1}	0.512×10^{-1}
False Abort	0.483×10^{-2}	0.484×10^{-2}	0.501×10^{-2}
Mission Success	0.94320	0.968717	0.968718

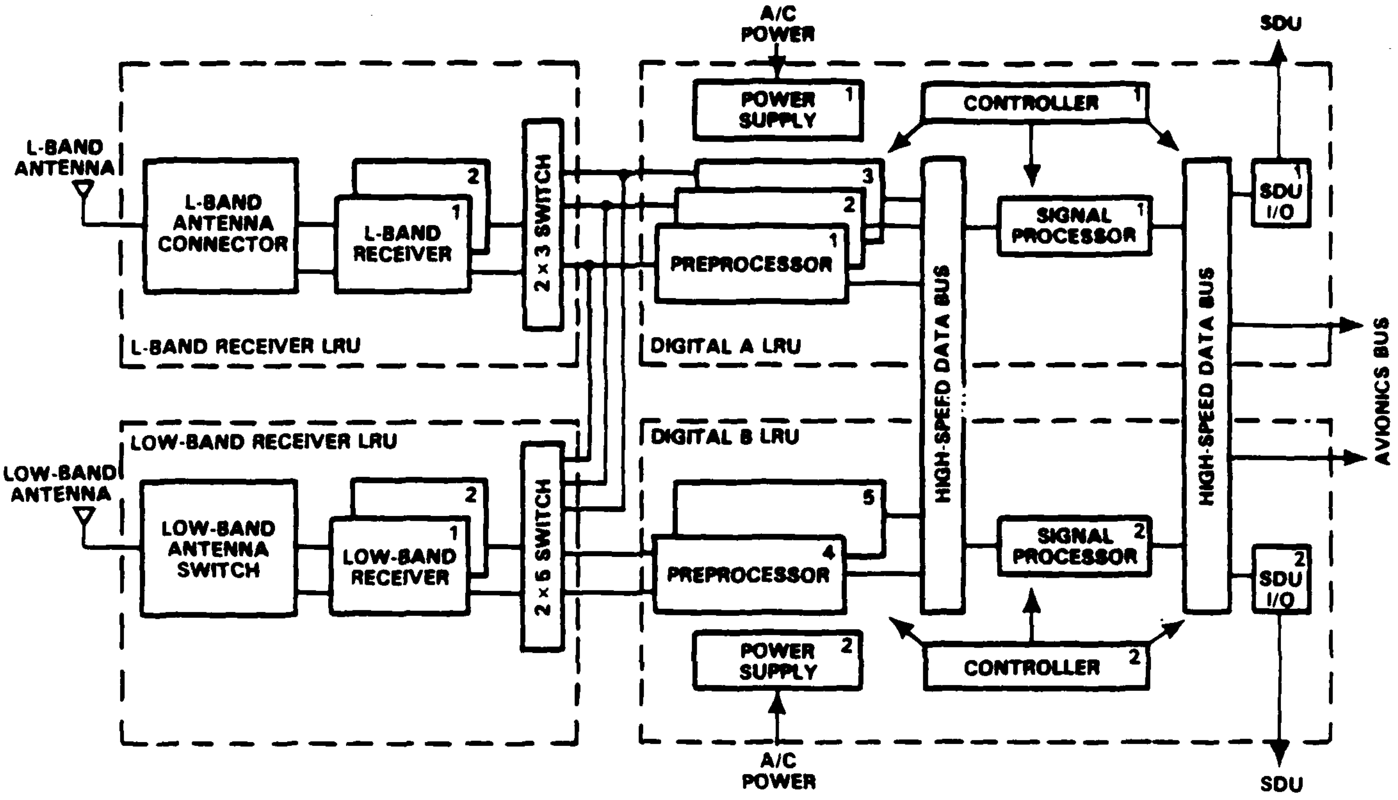


Figure 2. Example 2 Architecture.

Table 2. Example 2 Testability Results

Outcome	Lower bound	Upper bound
Correct Abort	0.4927×10^{-2}	0.4967×10^{-2}
Unknown Mission Failure	0.0166×10^{-1}	0.2078×10^{-1}
False Abort	0.6922×10^{-2}	0.7001×10^{-2}
Mission Success	0.96737	0.97137

that in this case the only two possible outcomes are false abort and mission success. As expected, the mission success probability was the same as in the previous case to the sixth decimal place.

2.3 Allocation and Reallocation of Critical Functions

The bounds on the probability of mission success presented above did not depend on the algorithm for allocating and reallocating critical functions. A good algorithm will result in a success probability closer to the upper bound than a poor algorithm. Note that any algorithm that selects an allocation that supports the mission (based on known failures) whenever possible will perform within the bounds presented in Section 2.2. Heuristic methods for selecting "good" allocation algorithms that minimize the effects of the imperfect monitor are discussed in this section.

Until the mission is aborted, a nondetected failure must occur to cause mission failure. False alarms can only cause mission failure in conjunction with nondetected failures, or if they lead to a mission abort. Hence, minimizing the effect of nondetected errors will be the primary consideration; minimizing the effect of false alarms will be a secondary consideration.

To minimize the probability of a nondetected failure, the allocation algorithm should use branches with the smallest nondetected failure rate possible. The algorithm should not reallocate unless forced to do so by a detected failure. If forced, the algorithm should allocate functions to new branches with the smallest nondetected failure rate possible. The reason for this can be seen from a simple example. Suppose there are two identical branches, either of which could be used to support a specific function. At some point during the mission, each branch has a probability of 0.1 of having incurred a nondetected failure. If only a single branch has been used to support the function up to that point, there is 10% chance of having unwittingly used a defective branch. If the algorithm switches to the other branch, the probability of having unwittingly used a defective branch jumps to 0.19, almost twice as high. The only way to avoid using a defective branch when the algorithm switches branches is if both branches are working. Thus, it is better to use as few branches as possible. In practice, there may be other reasons, such as non-interruptive Built-In Test procedures or resource balancing requirements, why the controller would reallocate functions.

Let η_o denote the nondetected failure rate of branches being used at time 0. The strategy for initially allocating functions should be to minimize η_o. If there are several possible allocations minimizing η_o, secondary considerations can be used to select among them. For example, among those that minimize η_o, one might select an allocation that minimizes $\alpha_o + \delta_o$, where α_o and δ_o are the total false alarm and detected failure rates of branches being used at time 0. This scheme would maximize the expected time until the monitor detects a failure in a branch being used and is forced to reallocate. However, minimizing η_o would be the primary objective; maximizing the time until reallocation would be secondary.

The following scheme is proposed for reallocating when "forced" by detected failures or false alarms. Let η_i denote the total nondetected failure rate of components used up to and including the ith reallocation. The algorithm should select each successive allocation to minimize η_i and break ties based on $\alpha_o + \delta_o$, as above. This can be repeated until the monitor believes that the critical functions can no longer be supported.

A similar concept can be applied after the monitor believes that the critical functions cannot be supported. If the mission is continued rather than aborted, the critical functions must be allocated to branches which are believed to be down. Such a scheme would require that the monitor compute the conditional probability that a branch is operational given that a failure indication has been received. These probabilities will depend on how the failure/detection process is modeled. Given these probabilities, the monitor should select branches with a minimum probability of being down.

3. INNOVATIVE REPAIR POLICIES

Traditionally, logistics support concepts have included the premise that all faults in mission-critical equipment must be repaired before a weapon system can be utilized. This premise may need to be discarded as innovative repair policies are considered to exploit the fault-tolerance characteristics of advanced systems. Deferred repair policies, whereby some or all noncritical repairs are deferred, offer the potential for increased availability and sustainability of fully mission capable systems.

The MIREM framework was used as a basis for evaluating the reliability and availability implications of deferred repair policies. After discussions with the ICNIA development contractors, four repair policies were defined:

1. Immediate Repair: repair any faults at the end of each mission.

2. Deferred Repair: repair only when a critical failure occurs.

3. Scheduled Maintenance: repair after a specified operating time or when a critical failure occurs.

4. Repair at Degraded Level: repair when the number of redundant components in some portion of the system falls below a specified level; these repairs include repairing when a critical failure occurs.

3.1 Repair Policy Analysis

Deferral of repair in fault-tolerant systems will impact bc th reliability, due to starting missions with fewer redundant components, and availability, due to the increased operating time without repair and the opportunity to perform several repairs simultaneously. Reliability will be measured in terms of average Mission Completion Success Probability (MCSP) for a fleet of systems operating under a given repair policy. The deferral of repairs results in missions being started in various degraded (but still mission capable) states, so that a single MCSP number does not apply.

Inherent availability will be calculated as the ratio of Mean Time Between Maintenance Actions (MTBMA) to MTBMA plus Mean Time To Repair (MTTR). MTBMA is defined as the mean operating time until system repair, starting with a fault-free system. MTTR refers to the time to repair the system by removing and replacing Line Replaceable Units (LRUs) or Line Replaceable Modules; logistics downtime is not included.

The example architecture of Figure 3 will be used to illustrate the analysis. The Repair at Degraded Level policy is defined in Table 3 in terms of the repair level in each pool of interchangeable resources. A scheduled maintenance interval of

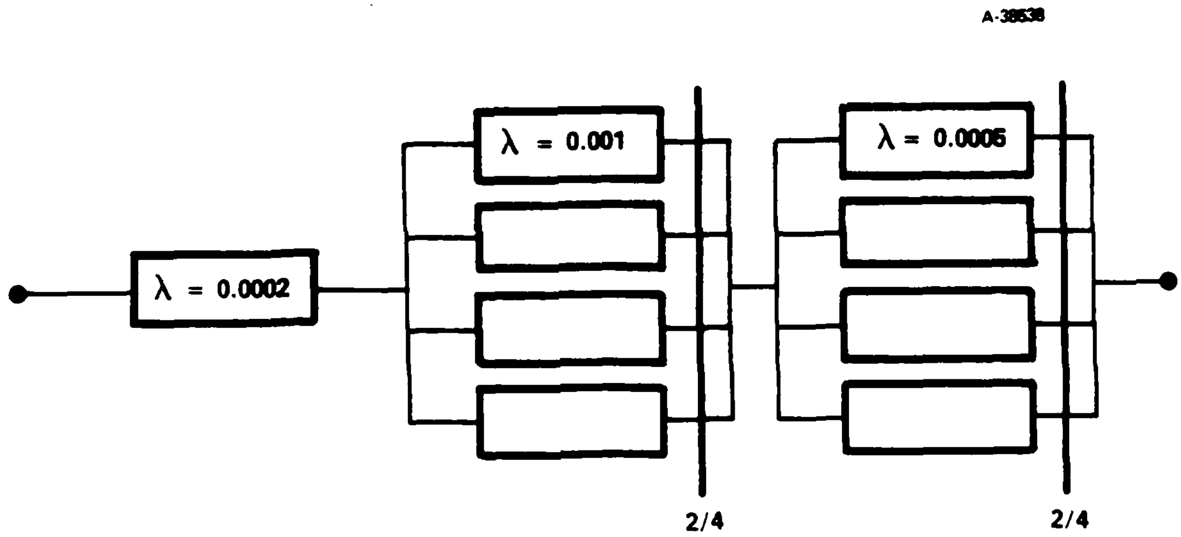


Figure 3. An Example Architecture.

Table 3. Repair Levels for Example Architecture

Pool	Number of branches	Number of branches needed to defer repair
1	1	1
2	4	3
3	4	2

100 hours, component MTTR of 2 hours, and mission length of 3 hours are assumed. Figure 4 shows the MIREM results for this example. Average MCSP, or equivalently, Mean Time Between Critical Failure, is highest for the immediate repair policy (0.9994) and lowest for the deferred repair policy (0.9964). These results reflect the poorer state of repair in which the deferred repair policy maintains the system. Conversely, availability is lowest for immediate repair (0.988) and highest for deferred repair (0.998).

The impact of scheduled maintenance will depend on the maintenance interval. In this example, deferring repairs for 100 hours had only a slight impact on reliability. The scheduled maintenance downtime is not counted in the availability measure unless repairs are performed.

The Repair at Degraded Level policy allows the logistician to optimize the repair decision against operational goals. For example, the repair levels shown in Table 3 are optimal (give the highest reliability) against an availability goal of 0.995.

3.2 IMIS Diagnostic Technology

One system that may help to exploit deferred repair policies is the Integrated Maintenance Information System (IMIS) being developed by the Air Force Human Resources Laboratory to provide an integrated source of automated maintenance information for the flightline technician. The IMIS information network is shown in Figure 5. The technician will possess a portable computer display which can be plugged into an aircraft maintenance panel, and which also has radio links to airborne systems and base maintenance computers. IMIS will display graphic technical instructions, analyze recorded flight data and aircraft historical data to provide diagnostic advice, and interrogate airborne systems. It will provide a means for the technician to receive work orders, report maintenance actions, order parts from supply, and receive computer-aided training. The maintenance workstation will allow the technician to exchange information with other base computer

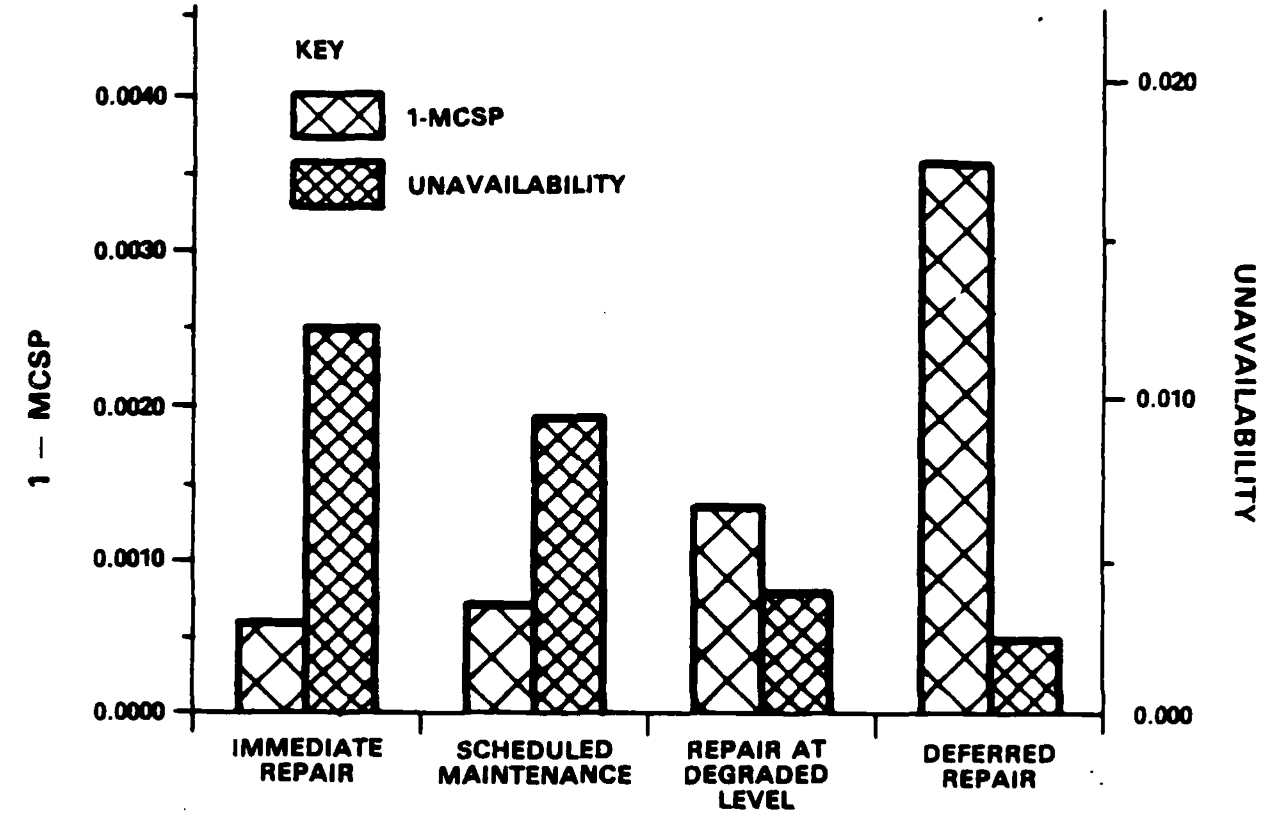


Figure 4. Reliability and Availability as a Function of Repair Policy.

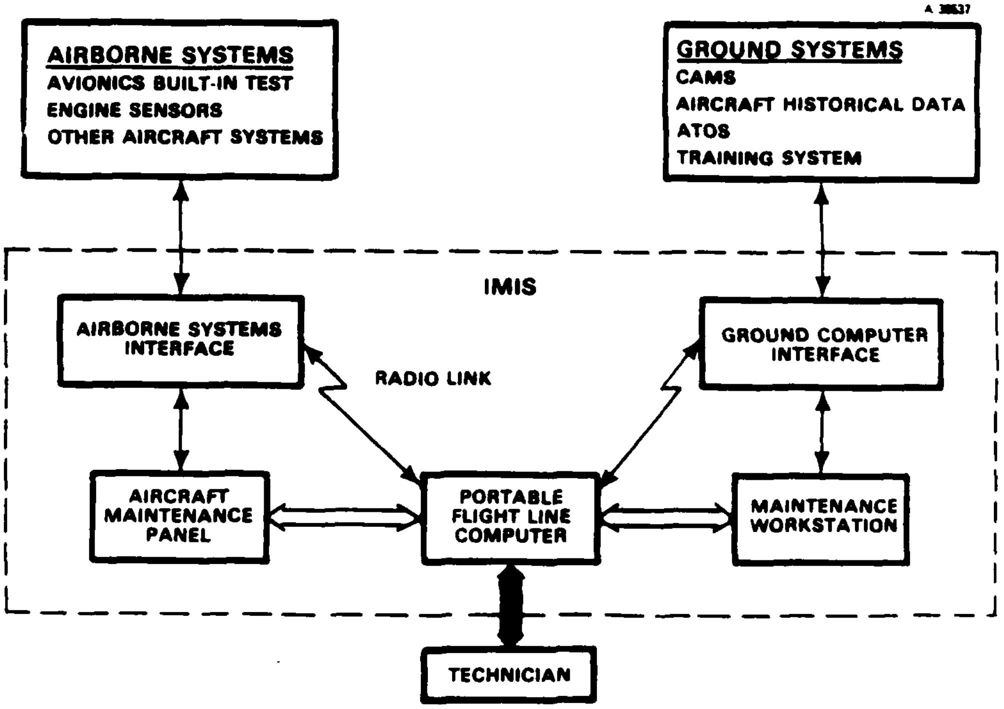


Figure 5. IMIS Information Network.

systems, such as the Core Automated Maintenance System (CAMS) and the Automated Technical Order System (ATOS).

One of the more sophisticated functions performed by IMIS will be diagnostics. Systems such as the Advanced Tactical Fighter will possess extensive on-board fault detection/isolation capability and graceful degradation. The ability of advanced systems to reconfigure, or self-repair, after a failure offers the potential for innovative repair concepts and complicates the decision of what to replace. IMIS will contain additional, independent fault isolation software and artificial intelligence techniques to provide diagnostic advice.

3.3 Computer-Aided Maintenance Decisions Using MIREM

As demonstrated in Section 3.1, MIREM now has the capability to evaluate the reliability and maintainability impacts of deferred repair policies and can be used interactively to construct a repair policy that achieves certain goals. Repair of non-critical failures may be deferred to achieve higher availability and more sorties. IMIS provides an environment in which these repair policies could be implemented. Determination of the policy requires reliability, maintainability, and operational requirement data that are not typically available to an on-board system. Data availability and computer resource requirements make a ground-based system, such as IMIS, preferable for determining and storing repair policies. Figure 6 illustrates how MIREM could be incorporated into IMIS. Repair policies would be developed by periodically running MIREM on the cognizant Air Logistics Center (ALC) computer, using Air Force-wide historical data. The repair policy would then be loaded into the IMIS portable computer diagnostics for the appropriate aircraft configuration and mission. When system status is read from the aircraft maintenance panel, the combination of healthy and failed modules would be looked up in a repair policy table and a recommendation made to the technician whether or not to repair the system before flying a specified mission.

It is recognized that this maintenance decision aid is a "policy" only in the sense of a repair or defer recommendation for every failure contingency. Other factors, such as operational priorities and availability of spares, will certainly influence the repair decision. Deferred repair policies constitute a major departure from current maintenance practices. Their institutionalization would require fundamental changes in the way that maintenance crews view their jobs.

4. CONCLUSIONS AND RECOMMENDATIONS

Algorithms have been developed to assess the impact of imperfect fault detection/isolation and innovative repair policies

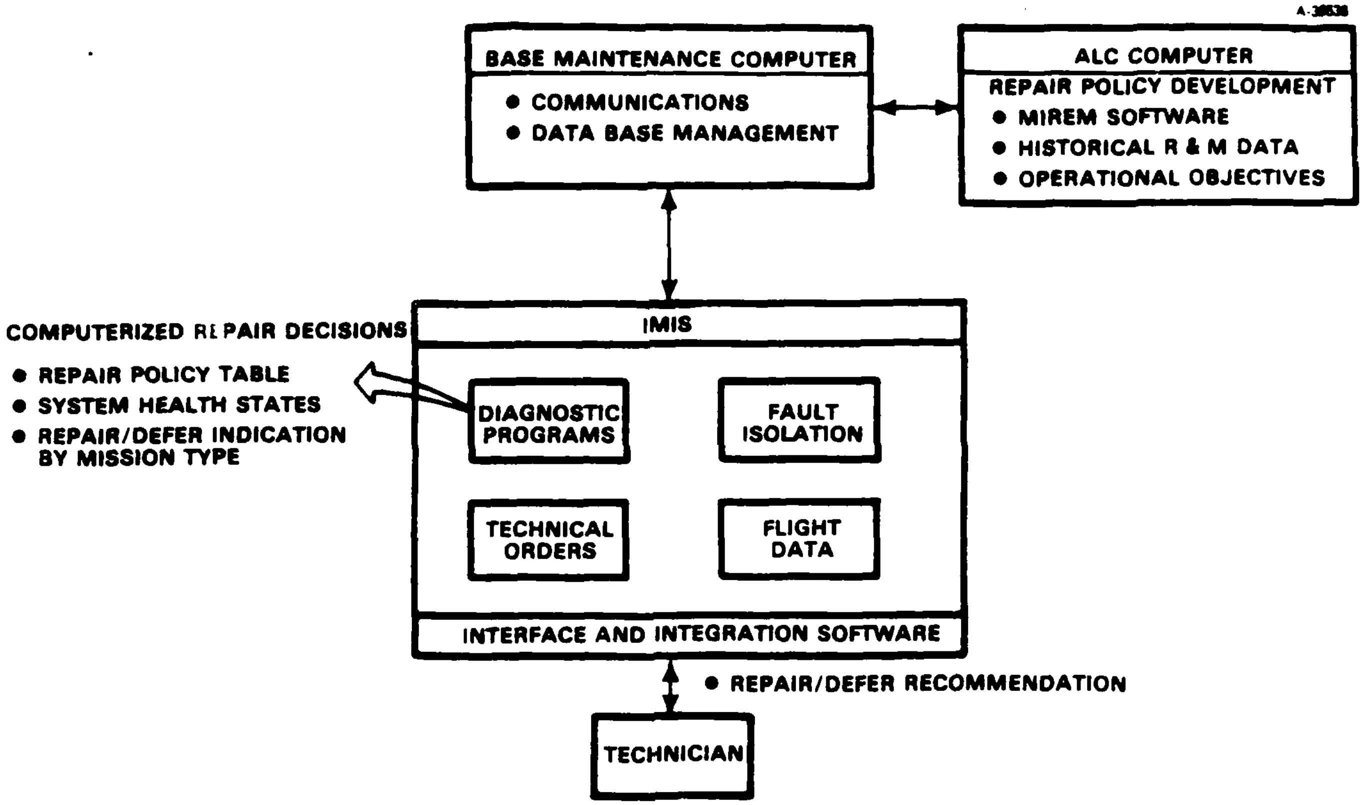


Figure 6. The Application of MIREM to IMIS.

on the reliability and availability of fault-tolerant systems. These algorithms apply to the class of systems modeled by MIREM, and provide valuable extensions to the MIREM methodology. MIREM now contains a fairly comprehensive treatment of hardware reliability. The model now captures enough factors so that it would be reasonable to use the model in a predictive mode (e.g., to predict performance during reliability testing), rather than as a tradeoff tool. However, accuracy of the results is still dependent on accuracy of the failure rate inputs.

Several insights were gained by applying these algorithms to test problems. In the testability area:

1. Mission reliability is more sensitive to undetected failures than to false alarms, particularly for highly fault-tolerant systems.

2. The number of false aborts and unknown mission failures (due to imperfect testing) can be greatly affected by the reallocation scheme that is used to manage fault tolerance.

In the repair policy area:

1. Deferral of repair of noncritical failures can greatly

extend the time between maintenance actions; however, for systems without single-point critical failures, the reliability penalty can be significant.

2. Scheduled maintenance policies offer simplicity and can effectively maintain systems at a high level of mission reliability.

3. A policy that repairs the system when it degrades below specified levels of redundancy offers the best tradeoff between reliability and availability; it is also the most difficult to implement.

It is recommended that MIREM3, which contains most of these algorithms, be tested on a system in the design process. The reasonableness of the results and the usability of the model should be evaluated. The ICNIA development contractors, who are already using MIREM, offer an excellent opportunity to have the new model accepted and used. Applications to failure modes, effects, and criticality analysis using the testability features, and to logistics support planning using the repair policy features, should be investigated.

Another issue that was identified during this research is the impact of the resouce allocation process on system reliability. The manner in which the system is reconfigured in response to faults or for other reasons will impact reliability through the mechanism of undetected faults. It is recommended that emphasis be placed on reconfiguration logic for reconfigurable systems, including the requirement that reliability impacts be addressed.

Finally, it is recommended that automated recommendations on whether to defer a repair be included as an IMIS function. As IMIS development continues, the MIREM integration issues that arise at the time should be addressed.

REFERENCES

Foley, R.D., & Suresh, S. (1984). Avionics reliability analysis. Report submitted to the Southeastern Center for Electrical Engineering Education and to the Air Force Human Resources Laboratory.

Foley, R.D., & Suresh, S. (1986). Avionics testability analysis. Report submitted to the Analytic Sciences Corporation.

Veatch, M.H., & Calvo, A.B. (1983). Reliability/Logistics Analysis Techniques for Fault-Tolerant Architectures, IEEE NAECON '83 Proceedings, pp. 675-681.

Veatch, M.H., Calvo, A.B., Myers, J.F., & McManus, J.C. (1985, November). Logistics engineering analysis techniques for fault-tolerant avionics systems (AFHRL-TR-84-60, AD-A161 981). Wright-Patterson AFB, OH: Logistics and Human Factors Division, Air Force Human Resources Laboratory.

Veatch, M.H., & Gates, R.K. (1986, November). Mission reliability model users guide (AFHRL-TR-86-35, AD-A175 235). Wright-Patterson AFB, OH: Logistics and Human Factors Division, Air Force Human Resources Laboratory.

APPENDIX A: IMPERFECT SWITCHING RELIABILITY COMPUTATIONS

A.1 Problem Statement

In this section, we depart somewhat from the description of Veatch, Calvo, Myers, and McManus (1985) in order to incorporate the concept of an imperfect monitor. Let 1 denote the working or good state and 0 the failed or bad state. Let $X_{ij}(t)$ be an ordered pair

$$X_{ij}(t) = [A_{ij}(t), B_{ij}(t)]$$

describing the actual and believed status of the jth branch in pool i at time t. $A_{ij}(t)$ is either 1 or 0, depending on whether the branch is actually up or down; and $B_{ij}(t)$ is either 1 or 0, depending on whether the monitor believes the brarch is up or down. With a perfect monitor, A(t) = B(t). Let X(t) be the matrix [A(t), B(t)]. We assume that initially all branches are believed to be and are actually working.

Each branch in pool i fails after an exponentially distributed length of time with parameter λ_i. These failures are detected with probability p_i. Thus, the rate at which detected failures occur is $\delta_i = \lambda_i p_i$, and the rate of nondetected failures is $\eta_i = \lambda_i(1 - p_i)$. In addition to failures, the length of time until the branch generates a false alarm is an expotentially distributed random variable with rate α_i. Thus, for each branch in pool i, δ_i is the rate at which detected failures occur, η_i is the rate at which nondetected failures occurs, and α_i is the rate at which false alarms occur. Assuming independence, $X_{ij}(t)$ is a Markov process with generator

$$\begin{array}{c c}
 & \begin{array}{cccc} (1,1) & (1,0) & (0,1) & (0,0) \end{array} \\
\begin{array}{c} (1,1) \\ (1,0) \\ (0,1) \\ (0,0) \end{array} &
\left[\begin{array}{cccc}
-(\alpha_i + \eta_i + \delta_i) & \alpha_i & \eta_i & \delta_i \\
0 & -(\eta_i + \delta_i) & 0 & (\eta_i + \delta_i) \\
0 & 0 & -(\alpha_i + \delta_i) & (\alpha_i + \delta_i) \\
0 & 0 & 0 & 0
\end{array}\right]
\end{array}$$

Note that all states are transient except for (0,0) which is absorbing. We assume that the functions $X_{ij}(t)$ are mutually independent processes. However, this does not completely describe the system since we also need to know how the functions are allocated.

Let L(t) denote the allocation of the functions to components. L(t) is a function of the believed states of the branches up to time t. Let $Y_A(t) = \phi[A(t), L(t)]$ be 1 if all critical functions are supported and 0 otherwise. The monitor believes the system is in state $Y_B(t) = \phi[B(t), L(t)]$.

The allocation of functions is not completely specified since we do not know the algorithm used to allocate functions. We assume only that the monitor will allocate the functions so that $Y_A(t) = 1$ if at all possible; any other objectives are secondary. Also, we assume that the monitor will abort the mission at time t_a when $Y_B(t)$ first equals 0 and the monitor believes a critical failure has occurred.

In addition to the processes $Y_A(t)$ and $Y_B(t)$, it will be convenient to introduce a third stochastic process $Y_C(t)$. $Y_C(t)$ is defined as follows: $Y_C(t)$ is 1 if there exists an allocation that supports all of the critical functions after neglecting nondetected failures. If the system is still incapable of supporting all of the critical functions, neglecting nondetected failures, then $Y_C(t)$ is 0.

Now we can define four outcomes for a mission of length t_m.

1. Mission Success M = (1,1): $Y_A(t) = Y_B(t) = 1$ for all $t \leq t_m$.

2. False Abort M = (1,0): $Y_B(t_m) = 0$, $Y_A(t) = 1$ for all $t \leq t_a$, and $Y_C(t_m) = 1$.

3. Unknown Mission Failure M = (0,1): $Y_A(t) = 0$ for some $t < \min\{t_a, t_m\}$.

4. Correct Abort M = (0,0): $Y_B(t_m) = Y_C(t_m) = 0$ and $Y_A(t) = 1$ for $t < t_a$.

It will become clear below that these outcomes are mutually exclusive and exhaustive. The motivation for the definition of mission success and unknown mission failure is fairly clear. If a mission is aborted without a prior mission failure, it is classified as a false or correct abort. Correct aborts include those missions that fail at the time they are aborted and those that would have failed before t_m if only detected failures are considered (fix all nondetected failures and remove all false alarms). Hence, missions aborted due to false alarms that would have been aborted later due to detected failures are considered correct aborts.

A.2 Reliability Bounds

Let T_N denote the time of the first nondetected error. The following lemma is easy to prove.

Lemma 1. The following hold:

a) $Y_C(t)$ and $Y_B(t)$ are nonincreasing,

b) $Y_A(t) \leq Y_C(t)$,

c) $Y_B(t) \leq Y_C(t)$,

d) If $T_N > t$ then $Y_A(t) = Y_C(t)$.

MIREM algorithms to determine the reliability of a system with a perfect monitor have been developed in Veatch et al. (1985) and Foley and Suresh (1984). Let $R_\beta(t)$ denote the reliability of a system under the assumption of a perfect monitor and a branch failure rate in pool i of β_i.

We are now in a position to determine the joint probability of $Q(t) = (Y_C(t), Y_B(t))$. This will be defined as $q_t(i,j)$.

Proposition 1. The following holds:

$$q_t(i,j) = \begin{cases} R_{\alpha+\delta}(t) & \text{if } (i,j) = (1,1), \\ R_\delta(t) - R_{\alpha+\delta}(t) & \text{if } (i,j) = (1,0), \\ 1 - R_\delta(t) & \text{if } (i,j) = (0,0), \\ 0 & \text{otherwise.} \end{cases}$$

Proof. From Lemma 1.a., we know that the only three cases with positive probability are (1,1), (1,0), (0,0). Now, $q_t(1,1)$ is simply $P\{Y_B(t) = 1\}$. The believed state behaves exactly the same as the earlier version of MIREM with the exception that false alarms are also treated as failures. Hence, $P\{Y_B(t) = 1\} = R_{\alpha+\delta}(t)$. Similarly, the probability of (0,0) is the same as

$$P\{Y_C(t) = 0\} = 1 - R_\delta(t).$$

The case (1,0) follows since the three terms must sum to 1.

Let η denote the total nondetected failure rate.

Proposition 2. The following hold:

a1) $P\{M = (1,1) | Q(t_m) = (1,1)\} = e^{-\eta t_m} + \int_0^{t_m} [1 - p^a_{def}(s)]\, \eta e^{-\eta s} ds$

a2) $P\{M = (0,1) | Q(t_m) = (1,1)\} = \int_0^{t_m} p^a_{def}(s)\, \eta e^{-\eta s} ds$

where $p^a_{def}(s)$ is the probability that a defective branch is used during (s,t_m) conditioned on $Q(t_m) = (1,1)$ and $T_N = s$.

b1) $P\{M = (1,0) | Q(t_m) = (1,0)\} = e^{-\eta t_m}$

$$+ \int_0^m [P\{Y_B(s) = 1 | Q(t_m) = (1,0)\}(1 - p^b_{def}(s))$$

$$+ P\{Y_B(s) = 0 | Q(t_m) = (1,0)\}]\, \eta e^{-\eta s} ds$$

b2) $P\{M = (0,1) | Q(t_m) = (1,0)\} = \int_0^{t_m} P\{Y_B(s) = 1 | Q(t_m) = (1,0)\}$

$$\eta e^{-\eta s} ds$$

where $p^b_{def}(s)$ is the probability that a defective branch is used during (s,t_a) conditioned on $Q(t_m) = (1,0)$ and $T_N = s < t_a$.

c1) $P\{M = (0,0) | Q(t_m) = (0,0)\} = e^{-\eta t_m}$

$$+ \int_0^{t_m} [P\{Y_B(s) = 1 | Q(t_m) = (0,0)\}(1 - p^c_{def}(s))$$

$$+ P\{Y_B(s) = 0 | Q(t_m) = (0,0)\}]\, \eta e^{-\eta s} ds$$

c2) $P\{M = (0,1) | Q(t_m) = (0,0)\} = \int_o^{t_m} P\{Y_B(s) = 1 | Q(t_m) = (0,0)\}$

$$p^c_{def}(s)\ \eta e^{-\eta s} ds$$

where $p^c_{def}(s)$ is the probability that a defective branch is used during (s,t_a) conditioned on $Q(t_m) = (0,0)$ and $T_N = s < t_a$.

Proof. In the right-hand side of each of the equations, we are conditioning on $s = T_N$, the first time of nondetected error. Note that T_N and Q are independent. In each case (a, b, and c), M equals either $Q(t_m)$ or $M = (0,1)$. Thus, we have

$$P\{M = Q(t_m) | Q(t_m)\} + P\{M = (0,1) | Q(t_m)\} = 1$$

In order for M to equal (0,1), we must have $T_N < t_m$, the monitor must believe the system is up at time T_N, and some branch containing a nondetected failure must be used before the mission is aborted. Substituting the appropriate probabilities gives (a2), (b2), and (c2).

Conversely, for M to equal $Q(t_m)$, we must have (A) $T_N \geq t_m$, (B) the monitor must believe the system is down at T_N, or (C) branches that contain nondetected failures but are not used before the mission is aborted. Expanding this condition logically as $A + A \cdot (B \cdot C + B)$ leads to (a1), (b1), and (c1).

We cannot directly compute some of the quantities in the right-hand side of Proposition 2. Instead, we will compute upper and lower bounds for those quantities.

Lemma 2. The following inequalities hold:

$$\underline{p} \leq p^a_{def}(s),\ p^b_{def}(s)\ p^c_{def}(s), \leq 1 \qquad \text{for } 0 \leq s \leq t_m$$

where $\underline{p}$ is defined as follows: Let $\underline{p}(i)$ denote the total nondetected failure rate of components used by the ith allocation divided by the total nondetected failure rate. Thus, $\underline{p}(i)$ represents the probability that a nondetected failure will occur in a component currently being used, given that the current allocation is i and that the nondetected failure just occurred. Then $\underline{p}$ is simply the minimum of $\underline{p}(i)$ over all possible allocations i.

It is difficult to tighten these bounds. In practice, one might expect $p^a_{def}(s)$, $p^b_{def}(s)$, and $p^c_{def}(s)$ to be close to $\underline{p}$ since one would expect the functions to be reallocated only if forced. However, the upper bound can be nearly obtained by continually reallocating the functions over all possible allocations. Such reallocation might occur to support the built-in test function. In order to define the quantities precisely, we would need the algorithm for allocating functions. We now state bounds for several other quantities that will be needed.

Lemma 3. The following inequalities hold:

$$P\{Y_B(s) = 1 | Q(t_m) = (1,0)\} \leq \min\{1, \frac{R_{\alpha+\delta}(s) - R_{\alpha+\delta}(t_m)}{R_\delta(t_m) - R_{\alpha+\delta}(t_m)}\}$$

$$P\{Y_B(s) = 0 | Q(t_m) = (1,0)\} \leq \min\{1, \frac{1 - R_{\alpha+\delta}(s)}{R_\delta(t_m) - R_{\alpha+\delta}(t_m)}\}$$

$$\frac{1 - R_\delta(s)}{1 - R_\delta(t_m)} \leq P\{Y_B(s) = 0 | Q(t_m) = (0,0)\} \leq \min\{1, \frac{1 - R_{\alpha+\delta}(s)}{1 - R_\delta(t_m)}\}.$$

Inserting the bounds from Lemmas 2 and 3 in Proposition 2, we obtain bounds on the mission outcome probabilities.

Proposition 3. The following hold:

$$p(1,1) \geq R_{\alpha+\delta}(t_m)\ e^{-\eta t_m}$$

$$p(1,1) \leq R_{\alpha+\delta}(t_m)\ [e^{-\eta t_m} + (1 - e^{-\eta t_m})(1 - \underline{p})]$$

$$p(1,0) \geq [R_\delta(t_m) - R_{\alpha+\delta}(t_m)]\ [e^{-\eta t_m}$$

$$+ \int_0^{t_m} \eta e^{-\eta s}(1 - \min(1, \frac{R_{\alpha+\delta}(s) - R_{\alpha+\delta}(t_m)}{R_\delta(t_m) - R_{\alpha+\delta}(t_m)}))ds]$$

$$p(1,0) \leq [R_\delta(t_m) - R_{\alpha+\delta}(t_m)]\ [e^{-\eta t_m}$$

$$+ \int_o^{t_m} \eta e^{-\eta s}[\min(1, \frac{R_{\alpha+\delta}(s) - R_{\alpha+\delta}(t_m)}{R_\delta(t_m) - R_{\alpha+\delta}(t_m)})(1-\underline{p})$$

$$+ \min(1, \frac{1 - R_{\alpha+\delta}(s)}{R_\delta(t_m) - R_{\alpha+\delta}t_m)})]ds]$$

$$p(0,0) \geq [1 - R_\delta(t_m)]\ [e^{-\eta t_m} + \int_o^{t_m} \eta e^{-\eta s}(\frac{1 - R_\delta(s)}{1 - R_\delta(t_m)})ds]$$

$$p(0,0) \leq [1 - R_\delta(t_m)]\ [e^{-\eta t_m}$$

$$+ \int_o^{t_m} \eta e^{-\eta s}[\frac{(R_\delta(s) - R_\delta(t_m)}{1 - R_\delta(t_m)})(1-\underline{p}) + \min(1, \frac{1 - R_{\alpha+\delta}(s)}{1 - R_\delta(t_m)})]ds]$$

$$p(0,1) \geq R_{\alpha+\delta}(t_m)\ (1 - e^{-\eta t_m})\ \underline{p}$$

$$+ [R_\delta(t_m) - R_{\alpha+\delta}(t_m)] \int_o^{t_m} \eta e^{-\eta s}$$

$$[(1 - \min(1, \frac{1 - R_{\alpha+\delta}(s)}{R_\delta(t_m) - R_{\alpha+\delta}(t_m)}))\ \underline{p}]ds$$

$$+ [1 - R_\delta(t_m)] \int_o^{t_m} \eta e^{-\eta s}[(1 - \min(1, \frac{1 - R_{\alpha+\delta}(s)}{1 - R_\delta(t_m)}))\ \underline{p}]ds$$

$$p(0,1) \leq R_{\alpha+\delta}(t_m)\ (1 - e^{-\eta t_m})$$

$$+ [R_\delta(t_m) - R_{\alpha+\delta}(t_m)]\int_o^{t_m} \eta e^{-\eta s} \min(1, \frac{R_{\alpha+\delta}(s) - R_{\alpha+\delta}(t_m)}{R_\delta(t_m) - R_{\alpha+\delta}(t_m)})ds$$

$$+ [1 - R_\delta(t_m)] \int_o^{t_m} \eta e^{-\eta s}(\frac{R_\delta(s) - R_\delta(t_m)}{1 - R(t_m)})ds$$

END

3-87

DTIC